

In the Claims:

Amend claims 16 and 21 as shown below in the entire set of pending claims. Underlines indicate insertions, and ~~strikeouts~~ or double brackets [[]] indicate deletions.

1 1. (Previously presented) An appliance application loading
2 system for a network environment, comprising:
3 a client;
4 a web application server communicating with the client within the
5 network environment and configured as a central location to store applications
6 and configuration settings for an appliance to enable configuration setups for a
7 plurality of appliances;
8 a plurality of network-based appliances communicably attached
9 with the web server within the network environment, wherein the network-
10 based appliances each include an embedded device having a non-volatile storage
11 device; and
12 a loading mechanism provided on the network-based appliance and
13 operative to download a first application to one of the appliances and a second
14 application to another of the appliances from the web application server upon
15 the occurrence of a power on/off cycle, wherein the loading mechanism is
16 configured to deliver the first application in a first, dedicated servlet to the one
17 appliance and deliver the second application in a second, dedicated servlet to the
18 another appliance where the second, dedicated servlet is unique from the first,
19 dedicated servlet, a first application header and a first universal resource locator
20 (URL) are stored on the non-volatile storage device for the one appliance and a
21 second application header and a second universal resource locator (URL) are
22 stored on the non-volatile storage device for the another appliance, first and
23 second application bodies are provided on the web server at a location
24 corresponding with the URL for the one appliance and the another appliance,
25 respectively, the URL being initialized to access the respective application body
26 when the loading mechanism detects the respective application header, the
27 loading mechanism is configured to automatically update one of the first

28 dedicated servlet and the second dedicated servlet responsive to a user locally
29 changing desired settings for one of the one appliance and the another
30 appliance, respectively, and one of the first and second application bodies is
31 configured to execute when one of a user and a selected one of the appliances
32 accesses the respective application body at the URL to download at least one of
33 the appliance packages, appliance configuration settings; and change
34 configuration of the respective appliance on the web application server.

1 2. (Original) The appliance application loading system of
2 claim 1 wherein the loading mechanism is provided at least in part by the client.

1 3-5. (Cancelled)

1 6. (Previously presented) The appliance application loading
2 system of claim 1 wherein each of the servlets is provided on the web server.

1 7. (Previously presented) The appliance application loading
2 system of claim 1 wherein each of the network-based appliances comprises an
3 embedded device, and the loading mechanism comprises a virtual machine.

1 8. (Previously presented) The appliance application loading
2 system of claim 1 wherein each of the network-based appliances uses the
3 loading device to download specific appliance configuration settings.

1 9. (Previously presented) The appliance application loading
2 system of claim 8 wherein each of the appliances comprises an embedded
3 device, and the loading mechanism comprises a program routine that copies an
4 application program into memory of the embedded device from the web server
5 for execution.

1 10. (Previously presented) A computer peripheral program
2 product, comprising:

a web application server configured as a central location to store applications and configuration settings for computer peripherals to enable configuration setups for a plurality of computer peripherals;

a network environment;

at least two computer peripherals; and

an application loader is configured to deliver a first application in a first, dedicated servlet to one computer peripheral and deliver a second application in a second, dedicated servlet to another computer peripheral where the second, dedicated servlet is unique from the first, dedicated servlet and to load a unique extendable architecture application to each of the at least two computer peripherals so as to enable versioning, updating, and remote configuration of the at least two computer peripherals via the web application server;

wherein the application loader associates a first application header of a first computer peripheral with a first application body of the web application server and a second application header of a second computer peripheral with a second application body of the web application server, wherein each of the applications includes a respective application header having identification information for the application and a uniform resource locator (URL) to the application body, the application body including one or more individual applications that can be loaded on the computer peripheral, the URL being initialized to access the application body when the application loader detects the application header, and the application loader is configured to automatically update one of the first dedicated servlet and the second dedicated servlet responsive to a user locally changing desired settings for one of the first computer peripheral and the second computer peripheral, respectively, and one of the first and second application bodies is configured to execute when one of a user and a selected one of the computer peripherals accesses the respective application body at the URL to download at least one of the appliance packages, and at least one of the appliance configuration settings; and to change configuration of the respective computer peripherals on the web application server.

1 11. (Previously presented) The computer peripheral program
2 product of claim 10 wherein each of the computer peripherals comprises a
3 virtual machine including a web client.

1 12. (Previously presented) The computer peripheral program
2 product of claim 10 wherein one of the computer peripherals comprises a
3 printer, and updating comprises configuring the one printer with a printer
4 application comprising a printer configuration state.

1 13. (Original) The computer peripheral program product of
2 claim 12 wherein the printer configuration state comprises user settings.

1 14. (Previously presented) The computer peripheral program
2 product of claim 12 wherein the printer configuration state is provided on one of
3 the servlets on the web application server that transfers applications and
4 settings to the printer in response to a power cycle that automatically updates
5 the applications and configuration settings for the printer.

1 15. (Cancelled)

1 16. (Currently amended) A method for updating applications to
2 embedded devices, comprising:

3 providing a plurality of network-based appliances each
4 communicably attached with a web application server, each of the appliances
5 having a loading mechanism to download an application to the appliance from
6 the server;

7 querying one of the appliances and another of the appliances with
8 the web server to determine presence of an application header for the one
9 appliance and the another appliance;

10 updating the one appliance with a first separate, dedicated
11 application from the server including a first set of user settings for the one
12 appliance and a second separate, dedicated application from the server including

13 a second set of user settings unique from the first set of user settings and for
14 the another appliance upon the occurrence of a power on/off cycle and upon
15 detecting the presence of the application header;
16 changing desired user settings for one of the one appliance and the
17 another appliance locally of the respective appliance; and
18 automatically updating one of a a ~~[[the]]~~ first dedicated servlet and a
19 ~~[[the]]~~ second dedicated servlet responsive to a user locally changing desired
20 user settings for one of the one appliance and the another appliance,
21 respectively.

1 17. (Original) The method of claim 16 wherein the appliance
2 comprises an embedded device, and updating comprises configuring the
3 embedded device with an application comprising an embedded device
4 configuration state.

1 18. (Original) The method of claim 17 wherein the embedded
2 device configuration state comprises user settings.

1 19. (Original) The method of claim 17 wherein the embedded
2 device configuration state comprises a servlet on the web application server that
3 is transferred to the embedded device in response to a power cycle that
4 automatically updates the applications and configuration settings for the
5 embedded device.

1 20. (Original) The method of claim 16 wherein a plurality of
2 appliances are communicably attached with the web application server each
3 with a dedicated one of the loading mechanism, wherein the web application
4 server stores appliance applications and configuration settings to enable plural
5 appliance configuration setup to version and update such applications.

1 21. (Currently amended) An appliance application loading system
2 for a network environment, comprising:
3 a client;

4 a server communicating with the client;
5 a plurality of network-based appliances communicably attached
6 with the server, the network-based appliances each including an embedded
7 device having a non-volatile storage device; and
8 a loading mechanism provided on the network-based appliance and
9 operative to deliver a [[the]] first application in a first, dedicated servlet to one
10 appliance and deliver a [[the]] second application in a second, dedicated servlet
11 to another appliance where the second, dedicated servlet is unique from the
12 first, dedicated servlet and to download a unique application to each of the
13 network-based appliances from the server upon the occurrence of a power
14 on/off cycle, wherein an application header and a universal resource locator
15 (URL) are stored on the non-volatile storage device of each appliance, and an
16 application body, having a servlet, is provided on the server at a location
17 corresponding with the URL for each appliance, the URL being initialized to
18 access the application body when the loading mechanism detects the application
19 header;
20 wherein servlet settings corresponding to each of the network-
21 based appliances are automatically updated via the loading mechanism to each
22 appliance if a user locally changes settings of the network-based appliance.

1 22. (Previously presented) The system of claim 21, wherein
2 each of the dedicated servlets is configured to have settings that are unique to
3 each of the network-based appliances.

1 23. (Previously presented) The appliance application loading
2 system of claim 1 further comprising a first network-based appliance and a
3 second network-based appliance, wherein the loading mechanism sets up the
4 one network-based appliance to have user settings for a first user who uses the
5 one network-based appliance locally, and the loading mechanism sets up the
6 another network-based appliance to have user settings for a second user who
7 utilizes the another network-based appliance locally, but remotely from the first
8 network-based appliance.

1 24. (Previously presented) The appliance application loading
2 system of claim 23 wherein the first network-based appliance comprises one
3 printer and the second network-based appliance comprises another printer, the
4 server is configured to update each unique application with fixes to software
5 viruses, and the loading mechanism is configured to download a first and second
6 updated unique application to the one printer and the another printer,
7 respectively.

1 25. (Previously presented) The appliance application loading
2 system of claim 24 wherein the loading mechanism utilizes the application body
3 comprising the respective, dedicated servlet, wherein the one printer is delivered
4 one servlet and the another printer is delivered another servlet, wherein the one
5 servlet comprises specific appropriate user settings for the one printer and the
6 another servlet comprises specific appropriate user settings for the another
7 printer.